PERGAMON Computers and Mathematics with Applications 38 (1999) 239–251

An International Journal
computers & mathematics
with applications

www.elsevier.nl/locate/camwa

Nonparametric Fuzzy Regression—k-NN and Kernel Smoothing Techniques

C.-B. CHENG AND E. S. LEE
Department of Industrial and Manufacturing Systems Engineering
Kansas State University, Manhattan, KS 66506, U.S.A.

(Received and accepted June 1998)

Abstract—Fuzzy regression without predefined functional form, or nonparametric fuzzy regression, is investigated. The two most basic nonparametric regression techniques in statistics, namely, k-nearest neighbor smoothing and kernel smoothing, are fuzzified and analyzed. Algorithms are proposed to obtain the best smoothing parameters based on the minimization of cross-validation criteria.

Keywords—Nonparametric fuzzy regression, k-NN smoothing, Kernel smoothing.

1. INTRODUCTION

Fuzzy regression with a predefined functional form has been investigated by many researchers. Unfortunately, for many practical problems, the functional form, or the relationship between the input and the output variables, cannot be obtained easily. Thus, several researchers have proposed the use of nonparametric, or model-free, regression. For example, by using the concepts of the neural network and back propagation, Tanaka and co-workers [1–4] proposed several nonparametric fuzzy regression approaches, where different types of weights and different error functions are used. A similar approach is also proposed by Fedrizzi *et al.* [5].

Based on the Sugeno fuzzy-rule model [6], Pokorný [7] formulated a fuzzy nonlinear regression approach, where crisp linear functions are replaced by possibilistic linear equations in the consequent section, and heuristic approach is used to identify the premise section. The training method used is similar to that used for the original Sugeno model.

In this paper, we propose to fuzzify and to analyze the two commonly used nonparametric regression techniques in statistical regression, namely, the k-nearest neighbor smoothing (k-NN) and the kernel smoothing techniques. As is well known, the performances of these statistical approaches are principally influenced by the neighborhood size or bandwidth. An algorithm based on the minimization of the cross-validation criterion is proposed to determine the bandwidth.

2. FUZZY REGRESSION

The function $f(X)$ is a mapping from X to Y, where $X = (x_1, \ldots, x_p)^{\mathrm{T}}$ is a p-dimension real vector representing the independent, or input, variables of the system. If Y is a fuzzy number, or

0898-1221/99/$ - see front matter. Typeset by $\mathcal{A}\mathcal{M}\mathcal{S}$-TEX
PII: S0898-1221(99)00198-4

the system structure is indefinite or fuzzy, then the general fuzzy regression model has the form

$$Y = f(X)\{+\}\epsilon, \tag{1}$$

or equivalently,

$$\epsilon = Y\{-\}f(X), \tag{2}$$

where ϵ is an observational error. Instead of being solely regarded as a random error with zero mean, ϵ may be considered as a fuzzy error due to the fuzzy structure of the system. It also may be a hybrid error which contains both fuzzy and random errors. $\{+\}$ or $\{-\}$ is an operator with the purpose of measuring the difference between the observed and the estimated outputs. The definition of $\{+\}/\{-\}$ depends on the fuzzy ranking methods used.

Y may be defined as an L-R type fuzzy number [8] with the following membership function:

$$\mu_Y(z) = \begin{cases} \mathcal{L}\left(\dfrac{y-z}{e^{\mathrm{L}}}\right), & z \le y, \quad e^{\mathrm{L}} \ge 0, \\ \mathcal{R}\left(\dfrac{z-y}{e^{\mathrm{R}}}\right), & z \ge y, \quad e^{\mathrm{R}} \ge 0, \end{cases} \tag{3}$$

where $\mathcal{L}$ and $\mathcal{R}$ denote the left and right reference functions, respectively, of the membership function Y. Y can also be represented as $Y = (y, e^{\mathrm{L}}, e^{\mathrm{R}})_{\mathrm{LR}}$, with y as the center or mode and $e^{\mathrm{L}}, e^{\mathrm{R}}$ as the left and right spreads of Y. The functions $\mathcal{L}$ and $\mathcal{R}$ can be defined differently under different assumptions. For a triangular fuzzy number, the L-R membership functions can be defined as

$$\mu_Y(z) = \begin{cases} 1 - \left(\dfrac{y-z}{e^{\mathrm{L}}}\right), & e^{\mathrm{L}} \ge (y-z) \ge 0, \\ 1 - \left(\dfrac{z-y}{e^{\mathrm{R}}}\right), & e^{\mathrm{R}} \ge (z-y) \ge 0, \\ 0, & \text{otherwise.} \end{cases} \tag{4}$$

Moreover, if $e^{\mathrm{L}} = e^{\mathrm{R}} = e$, a symmetric triangular fuzzy number is resulted, which can be written as $Y = (y, e)$. In this investigation, symmetric triangular fuzzy numbers will always be assumed.

For fuzzy linear regression, equation (1) reduces to

$$Y = A_0 + A_1 x_1 + A_2 x_2 + \cdots + A_p x_p, \tag{5}$$

where A_j, $j = 0, \ldots, p$, are fuzzy parameters. Based on (5), Tanaka [9] introduced the possibilistic fuzzy regression approach, where the parameters in model (5) are obtained by solving a linear programming problem with the objective of minimizing the system vagueness and subject to the constraint

$$[Y_i]_\alpha \subseteq \left[\widehat{Y}_i\right]_\alpha, \qquad i = 1, \ldots, N, \tag{6}$$

where $\widehat{Y}_i$ is the estimated value of the i^{th} observation Y_i and $[.]_\alpha$ is the α-level set. This restriction means that all the α-level set of the given samples should be included in the α-level set of the fuzzy model. Equation (6) is known as the inclusion condition.

Suppose N pairs of input-output data $(X_i, Y_i)_{i=1,\ldots,N}$ are available, with $Y_i = (y_i, e_i)$ and $X_i = (1, x_{1i}, \ldots, x_{pi})^{\mathrm{T}}$. The parameters A_j, $j = 0, \ldots, p$, in (5) can now be obtained by solving the following linear programming problem [10]:

$$\begin{aligned} &\min \sum_{i=1}^{N} (b_0 + b_1|x_{1i}| + \cdots + b_p|x_{pi}|), \\ &\text{s.t. } \mathbf{b} \ge 0; \\ &\quad \mathbf{a}^{\mathrm{T}} X_i + (1-\alpha)\mathbf{b}^{\mathrm{T}}|X_i| \ge y_i + (1-\alpha)e_i, \\ &-a^{\mathrm{T}} X_i + (1-\alpha)\mathbf{b}^{\mathrm{T}}|X_i| \ge -y_i + (1-\alpha)e_i, \qquad i = 1, \ldots, N. \end{aligned} \tag{7}$$

Let $\mathbf{A}$ be the vector A_j, $j = 0, 1, \ldots, p$, then $\mathbf{A} = (\mathbf{a}, \mathbf{b})$, where $\mathbf{a} = (a_0, \ldots, a_p)^\top$ and $\mathbf{b} = (b_0, \ldots, b_p)^\top$. In the above discussion, it was assumed that the relationship between the input and the output variables is known and is represented by the linear equation (5). However, in actual applications, this relationship is frequently unknown. Thus, nonparametric regression where the functional form is assumed unknown is needed.

In this paper, two extensively studied nonparametric regression techniques in statistics, k-nearest neighbor and kernel smoothing, are analyzed and extended to fuzzy regression. These two smoothing techniques are based on the concept of *local averaging*. In other words, the estimated value of the regression surface at point x_0 is the weighted average of the responses of the observations in the neighborhood of x_0. When a function is fairly smooth, local averaging can provide a good approximation of this function. Let X_i, $i = 1, 2, \ldots, N$ where the index is in ascending order, then the smoothing function based on local averaging can be represented as

$$S(x = X_i) = \underset{i-k \le j \le i+k}{\text{AVE}}(Z_j), \tag{8}$$

where AVE denotes the mean, median, or any weighted average, and Z_j is the observation at X_j and is a real number. The smoothing function $S(x)$, which is also called the smoother, is an estimator of the true function $f(x)$. Equation (8) gives an estimated response at the i^{th} observation and the parameter k defines the neighborhood or bandwidth of the smoothing function.

In order to extend equation (8) to fuzzy regression, the observations Z_js are replaced by fuzzy numbers Y_js and the following fuzzy arithmetic operations are needed.

The addition of two symmetric triangular fuzzy numbers $A = (a, c)$ and $B = (b, d)$:

$$A + B = (a, c) + (b, d) = (a + b, c + d).$$

The scalar multiplication of a symmetric triangular fuzzy number $A = (a, c)$:

$$r \cdot A = r \cdot (a, c) = (r \cdot a, |r| \cdot c),$$

where r is a scalar and $r \in \mathbf{R}$. Since Y_js are symmetric triangular fuzzy numbers and $Y_j = (y_j, e_j)$, $j = 1, \ldots, N$, the AVE function becomes

$$S(x = X_i) = \underset{i-k \le j \le i+k}{\widetilde{\text{AVE}}}(Y_j) = \left(\underset{i-k \le j \le i+k}{\text{AVE}}(y_j), \underset{i-k \le j \le i+k}{\text{AVE}}(e_j) \right), \tag{9}$$

where $\widetilde{\text{AVE}}$ is a fuzzy local averaging operator.

3. k-NEAREST NEIGHBOR SMOOTHING

The basic idea of smoothing is that if a function f is fairly smooth, then the observations made at and near x should contain information about the value of f at x. Thus, it should be possible to use local averaging of the data near x to construct an estimator for $f(x)$. This estimator for $f(x)$ is called the smoother. Several smoothing technique such as the k-nearest neighbor smoothing, kernel smoothing, spline and orthogonal series smoothing, etc., have been proposed. The first two techniques will be considered for fuzzy regression analysis in this and the next sections.

The k-nearest neighbor estimate is a weighted average in a varying neighborhood. The neighborhood is defined as the k-nearest neighbors of x in Euclidean distance. The k-NN weight sequence was introduced by Loftsgaarden and Quesenberry [11] in the related field of density estimation and has been used by Cover and Hart [12] for classification purposes. The k-NN smoother is defined as

$$\widehat{Y}_i = S(x = X_i) = \sum_{j=1}^{N} w_j(x) Y_j, \tag{10}$$

where $\widehat{Y}_i$ is the estimate of Y_i and $w_j(x)$, $j = 1, \ldots, N$, is the weight sequence defined through the set of indexes

$$J_x = \{j : X_j \text{ is one of the } k\text{-nearest observations to } x\}. \tag{11}$$

With this set of indexes for neighboring observations, the k-NN weight sequence is constructed as

$$w_j(x) = \begin{cases} \dfrac{1}{k}, & \text{if } j \in J_x, \\ 0, & \text{otherwise.} \end{cases} \tag{12}$$

Let the observations Y_i, $i = 1, \ldots, N$, where $Y_i = (y_i, e_i)$, be symmetric triangular fuzzy numbers, then according to equation (10) and by the use of fuzzy arithmetic, we have the regression equation

$$\widehat{Y}_i = S(x = X_i) = (\widehat{y}_i, \widehat{e}_i) = \left(\sum_{j=1}^{N} w_j(x) y_j, \sum_{j=1}^{N} w_j(x) e_j \right). \tag{13}$$

To illustrate the construction of weights, consider the following example.

Let $(X_i, Y_i)_{i=1,\ldots,5}$ be $(1, (5, 2)), (7, (12, 4)), (2, (3, 1)), (3, (1, 1)), (6, (4, 2))$, and we wish to compute the k-NN estimate for $x = X_4 = 3$. Assuming $k = 3$, then $J_{x=3} = \{1, 3, 4\}$ and

$$w_1(3) = \frac{1}{3}, \qquad w_2(3) = 0, \qquad w_3(3) = \frac{1}{3}, \qquad w_4(3) = \frac{1}{3}, \qquad w_5(3) = 0.$$

Thus, we have

$$\widehat{Y}_4 = \frac{1}{3} \cdot (5, 2) + 0 \cdot (12, 4) + \frac{1}{3} \cdot (3, 1) + \frac{1}{3} \cdot (1, 1) + 0 \cdot (4, 2) = \left(3, \frac{4}{3}\right).$$

Obviously, the inclusion condition (6), which was proposed by Tanaka [10] for fuzzy regression, cannot be satisfied by the using of the regression equation (13). However, this condition will be implicitly approximated in selecting the appropriate k value, which will be discussed later in connection with the discussion of smoothing parameter selection.

To assess the performance of the smoother, equation (13), a performance measure defined as $B = (1/N) \sum_{i=1}^{N} [Y_i - S(X_i)]^2$ is constructed. Notice that this performance measure is based on the calculation of the distance or difference between two fuzzy numbers, which are sets, not actual numbers. The problem is how to define this distance. Various fuzzy ranking approaches have been proposed in the literature to obtain this distance. The approach proposed by Chang and Lee [13] will be used. According to this approach, B can be expressed as

$$B = \frac{1}{N} \sum_{i=1}^{N} [y_i - \widehat{y}_i]^2. \tag{14}$$

3.1. Numerical Example

The following two functions are used to represent the relationship between the input and the output variables:

$$f_1(x) = \frac{x^2}{5} + 2e^{x/10} \tag{15}$$

and

$$f_2(x) = 10 + 5 \sin\left(0.025\pi(1 - x)^2\right), \tag{16}$$

and 100 pairs of sample data are generated for each function. These sample data are generated as follows.

SAMPLE 1. Generated for the function represented by equation (15): X_i, $i = 1, \ldots, 100$, are uniformly distributed within the interval $[0, 10]$; $Y_i = (y_i, e_i)$, $i = 1, \ldots, 100$, and in which $y_i = f_1(X_i) + \text{rand}[-0.5, 0.5]$, and $e_i = 1/4 f(X_i) + \text{rand}[0, 1]$, where $\text{rand}[l, u]$ denotes a random number between l and u.

SAMPLE 2. Generated for the function represented by equation (16): $X_i, i = 1, \ldots, 100$, are uniformly distributed within the interval $[0, 10]$; $Y_i = (y_i, e_i)$, $i = 1, \ldots, 100$, and in which $y_i = f_1(X_i) + \text{rand}[-0.5, 0.5]$, and $e_i = 1/3 f(X_i) + \text{rand}[0, 1]$.

The fuzzy regression results or smoothing results by using the fuzzy k-NN approach for Samples 1 and 2 are shown in Figures 1 and 2, respectively, with the smoothing parameter $k = 15$. The scatter of the observations or the sample data and the corresponding estimates are represented by the modes or the center value, lower limit, and upper limit.

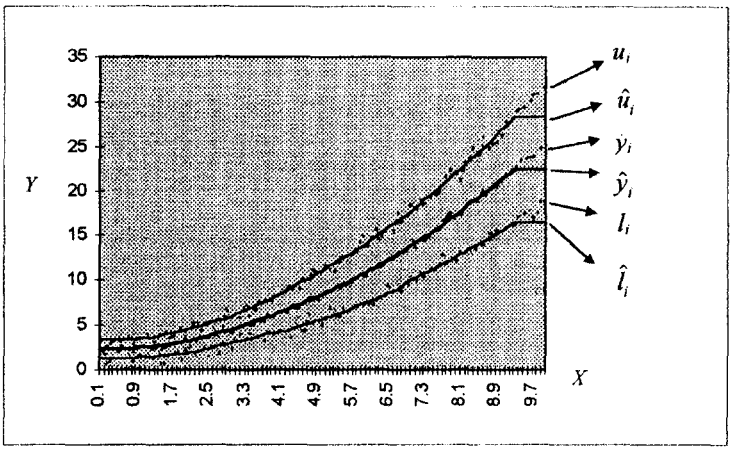

Figure 1. Regression results with Sample 1 data, k-NN smoothing.

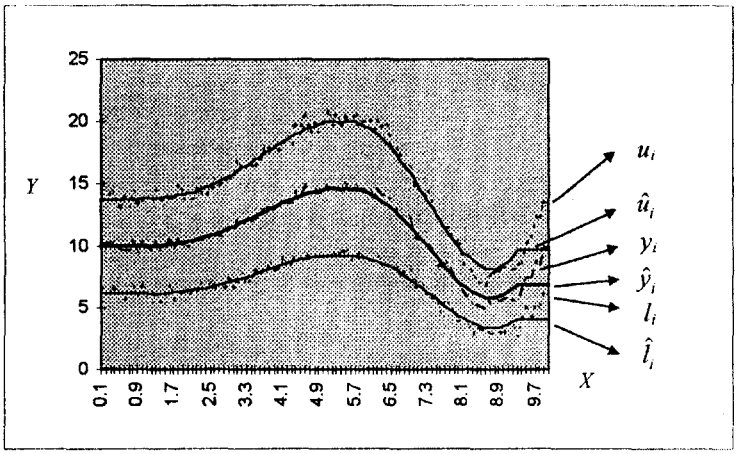

Figure 2. Regression results with Sample 2 data, k-NN smoothing.

3.2. Weight Adjustment

As shown in Figures 1 and 2, the results at the boundary are not as good as those far away from the boundary. This larger bias at the boundaries is due to the well-known boundary effect, which is caused by asymmetry at the neighborhood of the boundary. To reduce this effect, a simple adjustment is suggested in the weight sequence for points near the boundary. Triangular weights

centered at x, not uniform weights as constructed by (12), are used. Such a weight sequence is generated by

$$W_j(x) = \begin{cases} \dfrac{\left(\left|x - X^{(k)}\right| - |x - X_i|\right)}{\left|x - X^{(k)}\right|}, & \text{if } j \in J_x, \\ 0, & \text{otherwise,} \end{cases} \tag{17}$$

where $X^{(k)}$ denotes the k^{th} nearest neighbor of a boundary point x. The number of points which are classified as boundary points is determined subjectively. $w_j(x)$ is then obtained by normalizing $W_j(x)$

$$w_j(x) = \frac{W_j(x)}{\sum_t W_t(x)}.$$

With this weight adjustment, the same fuzzy regression problem with Samples 1 and 2 are again solved. The bias before and after weight adjustment is compared in Table 1. It can be seen that the bias is reduced by using this weights adjustment for the boundary points.

Table 1. The bias with and without weights adjustment, k-NN.

	Without Adjustment	With Adjustment
Sample 1 data	0.311	0.161
Sample 2 data	0.385	0.250

Although we only investigated problems with one-dimensional input, The approach can be easily extended to multidimensional problems. Since the determination of k-nearest neighbor only involves the calculation of Euclidean distances, the procedure for multidimensional input is essentially the same as that used for single dimensional input.

4. KERNEL SMOOTHING

A conceptually simple approach to represent the weight sequence in the local averaging method is to represent the weight distribution by a density function, which contains a scale parameter that adjusts the size and the form of the weights according to the location of the point with respect to the point of estimation x. This density function is known as the *kernel* function. Smoothing techniques based on this kind of weight representation are called kernel smoothing [14]. The construction of the kernel estimate differs from that used for the k-NN estimate. The k-nearest neighbor estimate is a weighted average in a varying neighborhood and the weights in a neighborhood are treated equally. The kernel estimate, $S(x)$, is defined as a weighted average of the response variable in a fixed neighborhood around x, determined in shape by the kernel function K and the bandwidth h.

The kernel estimate $S(x)$ is still represented by equation (10), but the weight sequence is generated by

$$w_j(x) = \frac{K_h(x - X_j)}{p_h(x)}, \tag{18}$$

where

$$p_h(x) = \sum_{j=1}^{N} K_h(x - X_j), \tag{19}$$

and in which

$$K_h(u) = \frac{1}{h} K\left(\frac{u}{h}\right) \tag{20}$$

is the kernel with scale factor h.

The shape of the kernel weights is determined by the kernel function K with the smoothing parameter h, which is called the bandwidth. The kernel function is a continuous, bounded, and symmetric real function which integrates to one,

$$\int K(v)\,dv = 1. \tag{21}$$

Although a variety of kernel functions can be constructed, practical and theoretical considerations limit the choice. For example, kernel functions that take on very small values can cause numerical underflow on a computer, thus one should avoid small values. One way to achieve this is to set any small values to zero [14]. Two commonly used kernel functions are analyzed and fuzzified in this investigation. The first is of the parabolic shape function

$$K_1(v) = \begin{cases} 0.75\left(1 - v^2\right), & \text{if } |v| \leq 1, \\ 0, & \text{otherwise,} \end{cases} \tag{22}$$

and the second is a Gaussian function

$$K_2(v) = (2\pi)^{-1/2} \exp\left(\frac{-v^2}{2}\right). \tag{23}$$

Notice that K_2 takes very small values when $|v|$ are relatively large. To avoid numerical underflow, set K_2=0, when $|v| > 3$.

The fuzzy regression equation for kernel smoothing remains the same as that for k-NN smoothing and is represented by equation (13). However, the weight sequence is now constructed by the use of equations (18)–(20). The inclusion condition (6) is still approximated by selecting an appropriate h value via a smoothing parameter selection procedure to be discussed later.

4.1. Numerical Example

The same two sets of data, Samples 1 and 2, are used for kernel smoothing. Only the results for $h = 0,75$ and $h = 0.25$ are shown in Figures 3 and 4, respectively. Since the same asymmetric neighborhood exists at the boundary points, the boundary effect still occurred in kernel smoothing. To reduce this boundary effect, weight adjustment for kernel smoothing was carried out by modifying (20) at the boundary points

$$K_h(u) = \left(1 - \left|\frac{u}{h}\right|\right) K\left(\frac{u}{h}\right), \qquad \text{if } K = K_1. \tag{24}$$

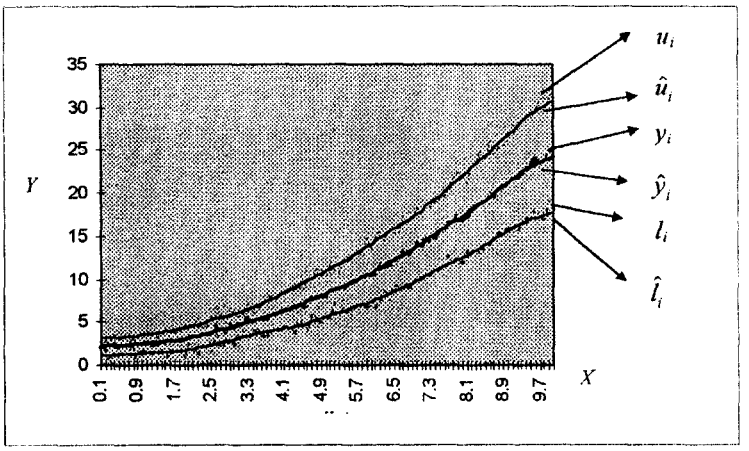

Figure 3. Regression results with Sample 1 data, K_1, $h = 0.75$.

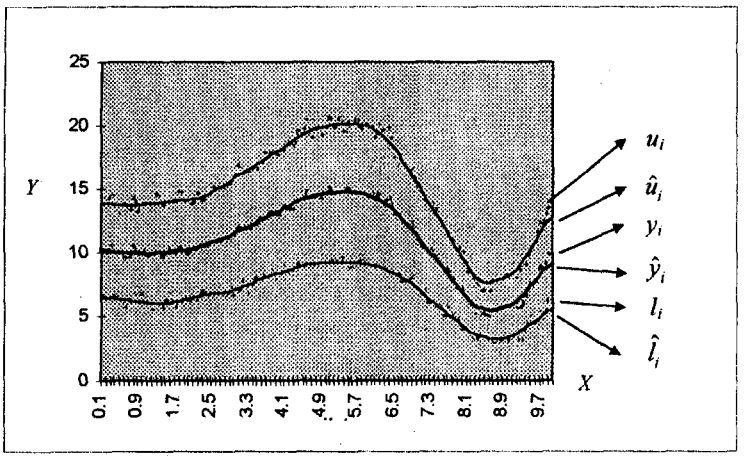

Figure 4. Regression results with Sample 2 data, K_2, $h = 0.25$.

Table 2. Bias measure for kernel smoothing with and without weight adjustment.

		Without Adjustment	With Adjustment
Sample 1	K_1	0.099	0.085
	K_2	0.080	0.076
Sample 2	K_1	0.142	0.108
	K_2	0.085	0.077

and

$$K_h(u) = \left(3 - \left|\frac{u}{h}\right|\right) K\left(\frac{u}{h}\right), \qquad \text{if } K = K_2. \tag{25}$$

Equations (24) and (25) increase the weights for points which are closer to the boundary. The bias measures for kernel smoothing with and without weights adjustment are listed in Table 2.

4.2. Multidimensional Input

In the above discussions, only the one-dimensional case was considered. For the multidimensional case with variables $X_i = (x_{i1}, \ldots, x_{in})^\top$, the following multidimensional product kernel function can be used:

$$K(v_1, \ldots, v_p) = \prod_{n=1}^{p} K(v_n). \tag{26}$$

The kernel weights are

$$w_j(x) = \frac{\prod_{n=1}^{p} K_h(x_n - x_{jn})}{p_h(x)}, \tag{27}$$

where K_h can be defined in a similar manner.

5. SMOOTHING PARAMETER SELECTION

Both approaches discussed in this investigation are essentially local averaging approaches. The most important aspect for these averaging techniques is to decide the size of the neighborhood to average. The regression error or bias can be reduced if a relatively small neighborhood size is used, but this will increase the regression noise. On the other hand, the regression noise can be reduced if a relatively large neighborhood size is used, but this will increase the regression error or

bias. In the extreme case, if the neighborhood size is the same as the sample size, the estimation becomes an average of all the sample data, which is certainly not desirable and would produce a very large bias. Thus, there is a trade-off between using a too large and a too small neighborhood. Obtaining the best choice between this trade-off is known as the smoothing parameter selection problem.

The k-NN smoothing parameter is the neighborhood size k. As for kernel smoothing, the effective weight function is determined by the kernel K and the bandwidth h. However, Härdle [14] argued that the weight mainly depends on the smoothing parameter h. Thus, the smoothing parameter selection problem will be concerned with the determination of the values of k and h only.

The Cross-Validation (CV) technique [15] will be used to assess the regression results and the optimal bandwidth. The best compromised value for k or h will be obtained by minimizing the cross-validation criterion.

The basic idea of the cross-validation approach is to divide the sample size N into a "construction" subsample with a sample size $N-1$ and a "validation" subsample with a sample size 1. The process must consider the division in all N possible ways. The approach is called leave-one-out cross-validation. According to Stone [15], the CV criterion is

$$CV(b) = \frac{1}{N}\sum_{i=1}^{N} L\left[Y_i, \widehat{f}_b\left(X_i, O_{\setminus i}\right)\right], \tag{28}$$

where b is the smoothing parameter; L is a loss function defined on the observed response Y_i and the corresponding estimated response $\widehat{f}_b(X_i, O_{\setminus i})$, which is estimated from a set of sample data O, excluding the i^{th} observation. The smoothing parameter b corresponds k for k-NN smoothing and the bandwidth h for kernel smoothing. The response function $\widehat{f}_b(X_i, O_{\setminus i})$ is defined as

$$\widehat{f}_b\left(X_i, O_{\setminus i}\right) = \sum_{j\neq i, j=1}^{N} w_j(X_i) Y_j, \tag{29}$$

where $w_j(X_i)$ is a weight function with respect to X_i.

In order to form a proper cross-validation formula, the loss function L must be considered carefully. At least the following two aspects should be considered in this formulation the measure of difference and the measure of inclusion. The measure of difference is used to evaluate the bias between the observations and their corresponding estimates so that the performance of the smoother can be assessed. The measure of inclusion is needed so that the solution can satisfy the inclusion condition (6) as proposed by Tanaka and co-workers. The details of these two measures are discussed in the following.

(i) The measure of difference. The difference between the estimates and actual can be expressed as

$$D_i(b) = \left[Y_i - \widehat{f}_b\left(X_i, O_{\setminus i}\right)\right]^2. \tag{30}$$

Since this difference is between fuzzy numbers which are sets, not crisp numbers, fuzzy ranking method must used. There are many different fuzzy ranking methods for measuring the difference between two or more fuzzy numbers. Chang and Lee [16] made an extensive survey about fuzzy ranking methods. In this investigation, the method of Chang and Lee [13], which is based on the concept of overall existence, will be used. An overall existence measure of an L-R type fuzzy number A is defined as

$$OM(A) = \int_0^1 \varpi(\nu)\left[\chi_1(\nu)\mu_{A_{\mathrm{L}}}^{-1}(\nu) + \chi_2(\nu)\mu_{A_{\mathrm{R}}}^{-1}(\nu)\right] d\nu, \tag{31}$$

where ν is the membership function value; $\mu_{A_L}^{-1}(\nu)$ and $\mu_{A_R}^{-1}(\nu)$ are the lower and upper limits of the w-level cut of fuzzy number A; and $\varpi(\nu)$, $\chi_1(\nu)$, and $\chi_2(\nu)$ are weights measures, which must be determined subjectively by the decision maker. For simplicity, we let

$$\varpi(\nu) = 1, \qquad \chi_1(\nu) = \chi_2(\nu) = \frac{1}{2}, \qquad \text{for all } \nu \in (0, 1]. \tag{32}$$

Therefore, if $A = (a, c^{\mathrm{L}}, c^{\mathrm{R}})_{\mathrm{LR}}$ is a triangular fuzzy number, we have

$$OM(A) = \frac{4a - c^{\mathrm{L}} + c^{\mathrm{R}}}{4}. \tag{33}$$

Thus, $D_i(b)$ is calculated by using equation (33).

(ii) The measure of inclusion. In order to approximate condition (6), a penalty function is constructed as

$$C_i(b) = P\left(\left[\widehat{f}_b\left(X_i, O_{\setminus i}\right)\right]_\alpha^{\mathrm{L}} - [Y_i]_\alpha^{\mathrm{L}}\right) + Q\left([Y_i]_\alpha^{\mathrm{R}} - \left[\widehat{f}_b\left(X_i, O_{\setminus i}\right)\right]_\alpha^{\mathrm{R}}\right), \tag{34}$$

where $[.]_\alpha^{\mathrm{L}}$ and $[.]_\alpha^{\mathrm{R}}$ denote the lower and upper limits of the α-level cut of a fuzzy number. The level α must be determined subjectively by the decision maker. P and Q are penalty terms and are determined by

$$P = \begin{cases} 1, & \text{if } [Y_i]_\alpha^{\mathrm{L}} \le \left[\widehat{f}_b\left(X_i, O_{\setminus i}\right)\right]_\alpha^{\mathrm{L}}, \\ 0, & \text{otherwise;} \end{cases} \tag{35}$$

and

$$Q = \begin{cases} 1, & \text{if } [Y_i]_\alpha^{\mathrm{R}} \ge \left[\widehat{f}_b\left(X_i, O_{\setminus i}\right)\right]_\alpha^{\mathrm{R}}, \\ 0, & \text{otherwise.} \end{cases} \tag{36}$$

Thus, violation of condition (6) will be penalized according to equations (35) and (36).

The loss function is essentially composed of the two measures of differences. By the use of these two measures of differences, which were obtained above, the cross-validation criterion becomes

$$CV(b) = \frac{1}{N}\sum_{i=1}^{N} D_i(b) + C_i(b). \tag{37}$$

Our objective is to find a smoothing parameter b which minimizes the above CV criterion, equation (37).

The minimization of (37) to obtain the best smoothing parameter can be carried out as follows.

Start with a small initial value b_0, and then increase the b_0 value gradually. For a given current value of b, the smoothing procedure, either for k-NN or for kernel smoothing, is executed in a leave-one-out manner, and the corresponding CV(b) value is calculated. This process is repeated with different bs. The optimal b^* is the one which produces the minimal CV(b) among all the bs. Finally, this smoothing procedure is carried for all N pairs of the data using b^* as a smoothing parameter. This algorithm produced a local minimum. The detailed algorithm for searching the best smoothing parameter is as follows.

5.1. Smoothing Parameter Selection Algorithm (SPSA)

Step 1. Initialization
set α value;
choose initial bandwidth b_0;
set the search range r,

Table 3. Solution sequence, k-NN.

	Sample 1	Sample 2
k	CV	CV
10	3.958	2.520
11	3.055	2.178
12	2.893	1.877
13	2.264	1.704
14	2.235	1.534
15	1.769	1.446
16	1.809	1.346
17	1.443	1.294
18	1.524	1.240
19	1.212	1.226
20	1.321	1.209*
21	1.070	1.203
22	1.203	1.220
23	0.998	1.226
24	1.147	1.272
25	0.972*	1.282
26	1.135	1.348
27	0.999	1.353
28	1.171	1.440
29	1.046	1.444
30	1.234	1.548

* local minimum

CV cross-validation value α is set to 0.1.

Table 4. Solution sequence, kernel smoothing.

	Sample 1	Sample 2
h	CV	CV
1	2.996	1.872
1.1	2.555	1.636
1.2	2.256	1.473
1.3	1.944	1.354
1.4	1.749	1.274
1.5	1.543	1.243
1.6	1.397	1.227
1.7	1.287	1.224*
1.8	1.169	1.262
1.9	1.097	1.294
2	1.015	1.367
2.1	0.963	1.442
2.2	0.925	1.525
2.3	0.889	1.632
2.4	0.872*	1.742
2.5	0.876	1.858
2.6	0.893	1.984
2.7	0.923	2.120
2.8	0.954	2.260
2.9	0.998	2.405
3	1.049	2.558

let the increment $d = b_0/r$;

set iteration counter iter $\leftarrow 1$.

Step 2. $b = (b_0 - (b_0/2) + (\text{iter} - 1) \times d$.

Call subroutine of smoother (k-NN or kernel smoothing).

Call subroutine of cross-validation, and returning CV(b).

Step 3. If (iter $> r + 1$) then go to Step 4;

otherwise, iter $\leftarrow$ iter $+ 1$, and goto Step 2.

Step 4. $b* = \text{argmin}\{\text{CV}(b)\}$.

Stop.

The initial bandwidth b_0 must be well chosen so that the local minimum can be reached quickly. For k-NN smoothing, the traditional regression analysis literature [14] suggests that for balancing variance and squared bias, k should be proportional to $N^{4/5}$. The initial bandwidth for k-NN

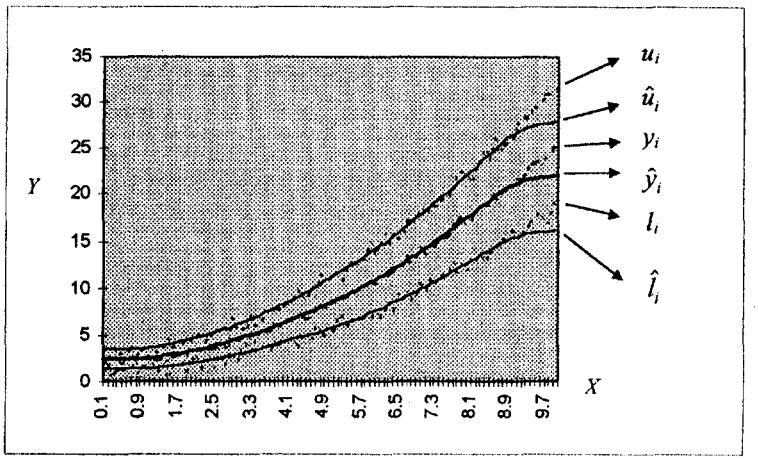

Figure 5. Regression results after parameter selection, k-NN.

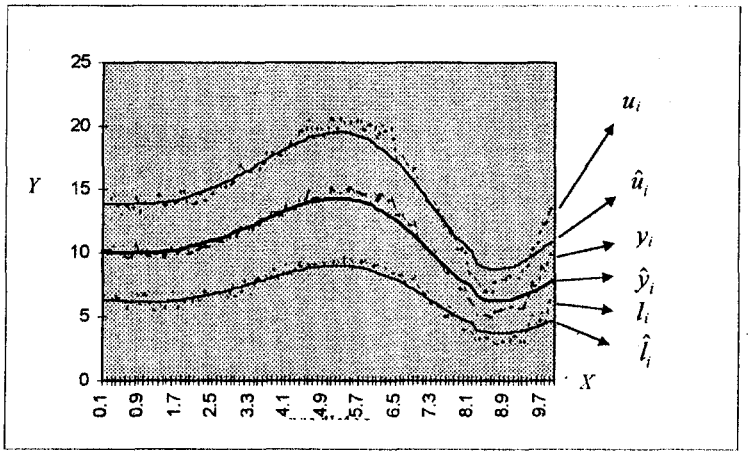

Figure 6. Regression results after parameter selection, kernel smoothing.

smoothing is chosen subjectively based on this suggestion. To cover about the same amount of observations in kernel smoothing, the bandwidth h can be determined as follows.

(i) Using kernel function K_1, if X_i, $i = 1, \ldots, N$, is uniformly distributed within a certain interval, we can estimate $S(X_i)$ with about k number of observations by letting

$$h = \frac{gk}{2N},$$

where $g = \max_i\{X_i\} - \min_i\{X_i\}$.

(ii) Using kernel function K_2, to have the same effect as in (i), let

$$h = \frac{gk}{6N}.$$

5.2. Numerical Examples

By using the same data generated before, this minimization problem is solved. Tables 3 and 4 show the sequences of solutions when the Proposed Smoothing Parameter Selection Algorithm (SPSA) is used for the selection of smoothing parameter b for either k-NN or kernel smoothing. Figure 5 and 6 show the regression results by using the best smoothing parameters obtained in Tables 3 and 4.

6. DISCUSSIONS

From Figures 1 and 3, we can see that the boundary effect for k-NN is more severe than that for kernel smoothing. The reason is that k-NN uses fixed number of observations for each estimate so the averaging procedure near the boundary involves more points. However, no conclusions can be made concerning which smoothing method is superior. For problems with sparse observations or sample, k-NN can always be used to obtain the estimation. But, due to the use of uniform weight sequence, the k-NN estimates are also represented by a rougher curve.

Since k-NN and kernel smoothing are nonparametric methods, they avoid some deficiencies of parametric fuzzy regression. Clemins [17] pointed out that Tanaka's LP fuzzy regression model often produces crisp coefficients. Chang and Lee [18] discussed the problem about the conflicting trades between the spread and the center line in the Tanaka model. These problems are avoided in the present approach.

In general, k-NN and kernel smoothing are simple and easy to implement. However, the problem of sparse data may reduce the effectiveness of the approaches. For example, for problems with high dimensional input variables, even a sample size of $N \leq 1000$ are surprisingly sparsely distributed. Thus, the number of observations in a neighborhood is frequently not enough to produce a good estimate.

REFERENCES

1. H. Ishibuchi and H. Tanaka, Fuzzy regression analysis using neural networks, *Fuzzy Sets and Systems* **50**, 257–265, (1992).
2. H. Ishibuchi and H. Tanaka, An architecture of neural networks with interval weights and its application to fuzzy regression analysis, *Fuzzy Sets and Systems* **57**, 27–39, (1993).
3. H. Ishibuchi and H. Tanaka, Fuzzy neural networks with fuzzy weights and fuzzy biases, In *Proceedings of 1993 IEEE International Conference on Neural Networks*, pp. 1650–1655, San Francisco, (March 28–April 1, 1993).
4. A. Miyazaki, K. Kwon, H. Ishibuchi and H. Tanaka, Fuzzy regression analysis by fuzzy neural networks and its application, In *Proceedings of 1994 IEEE International Conference of Fuzzy Systems*, pp. 52–57, Orlando, (June 26–29, 1994).
5. M. Fedrizzi and R.A.M. Pereira, Emulating fuzzy mappings with a neural network architecture, In *Proceedings of 1995 IEEE International Conference on Neural Networks*, pp. 251–254, Perth, Australia, (Nov. 27–Dec. 1, 1995).
6. T. Takagi and M. Sugeno, Fuzzy identification of systems and its application to modeling and control, *IEEE Transactions on Systems, Man, and Cybernetics* **15** (1), 116–132, (1985).
7. M. Pokorný, Fuzzy nonlinear regression method for intensification of the object's vagueness representation, In *Proceedings of 1995 IEEE International Conference of Fuzzy Systems*, pp. 2051–2055, Yokohama, Japan, (March 20–24, 1995).
8. D. Dubois and H. Prade, *Fuzzy Sets and Systems Theory and Applications*, Academic Press, New York, (1980).
9. H. Tanaka, Fuzzy data analysis by possibilistic linear models, *Fuzzy Sets and Systems* **24**, 363–375, (1987).
10. H. Tanaka, I. Hayashi and J. Watada, Possibilistic linear regression analysis for fuzzy data, *European Journal of Operational Research* **40**, 389–396, (1989).
11. D.O. Loftsgaarden and G.P. Quesenberry, A nonparametric estimate of a multivariate density function, *Annals of Mathematical Statistics* **36**, 1049–1051, (1965).
12. T.M. Cover and P.E. Hart, Nearest neighbor pattern classification, *IEEE Transactions on Information Theory* **13**, 21–27, (1967).
13. P.-T. Chang and E.S. Lee, Ranking of fuzzy sets based on the concept of existence, *Computers Math. Applic.* **27** (9/10), 11–21, (1994).
14. W. Härdle, *Applied Nonparametric Regression*, Cambridge University Press, New York, (1990).
15. M. Stone, Cross-validatory choice and assessment of statistical predictions, *Journal of the Royal Statistical Society* **36** (Series B), 111–147, (1974).
16. P.-T. Chang, Fuzzy regression analysis, Ph.D. Dissertation, Industrial and Manufacturing System Engineering Dept., Kansas State University, (1994).
17. A. Celmins, Least squares model fitting to fuzzy vector data, *Fuzzy Sets and Systems* **22**, 245–269, (1987).
18. P.-T. Chang and E.S. Lee, Fuzzy linear regression with spreads unrestricted in sign, *Computers Math. Applic.* **28** (4), 61–70, (1994).